LATERAL BUCKLING OF BEAMS WITHOUT WARPING RIGIDITY

Robert Edward Brown

NAVAL POSTGRADUATE SCHOOL

Monterey, California

THESIS

LATERAL BUCKLING OF BEAMS WITHOUT
WARPING RIGIDITY

by

Robert Edward Brown

June 1978

Thesis Advisor: J. E. Brock

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) Lateral Buckling of Beams Without Warping Rigidity		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis; June 1978
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Robert Edward Brown		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Naval Postgraduate School Monterey, California 93940		12. REPORT DATE June 1978
		13. NUMBER OF PAGES 58
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office) Naval Postgraduate School Monterey, California 93940		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Lateral buckling; lateral instability, buckling of beams.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A procedure is described to determine lateral buckling loads for initially straight beams. Loading and beam geometry may vary along the length. Warping rigidity is not considered. End conditions of considerable generality may be treated. The algorithm depends on solving a convergent sequence of sixteenth order eigenproblems. A computer program implementing this procedure has been developed and is

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
(Page 1) S/N 0102-014-6601 |

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

described herein.

Approved for public release; distribution unlimited.

Lateral Buckling of Beams Without Warping Rigidity

by

Robert Edward Brown
Lieutenant Commander, United States Navy
B.S., United States Naval Academy, 1968

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
June 1978

ABSTRACT

A procedure is described to determine lateral buckling loads for initially straight beams. Loading and beam geometry may vary along the length. Warping rigidity is not considered. End conditions of considerable generality may be treated. The algorithm depends on solving a convergent sequence of sixteenth order eigenproblems. A computer program implementing this procedure has been developed and is described herein.

TABLE OF CONTENTS

ACKNOWLEDGEMENT

The author wishes to express his appreciation to Dr. John E. Brock, Professor of Mechanical Engineering, for his invaluable guidance and advice. Without his inspiration and patient assistance this work could not have been accomplished.

II. PROBLEM SPECIFICATIONS

The coordinate system used is a right-handed orthogonal x,y,z system with positive senses as indicated in figure 1. The longitudinal centroidal axis of the beam lies along the +z direction. The principal axes of a normal cross section are originally in the x and y directions but after torsional displacement (rotation) they are in the ξ and η directions, cf. figure 2 (page 10).

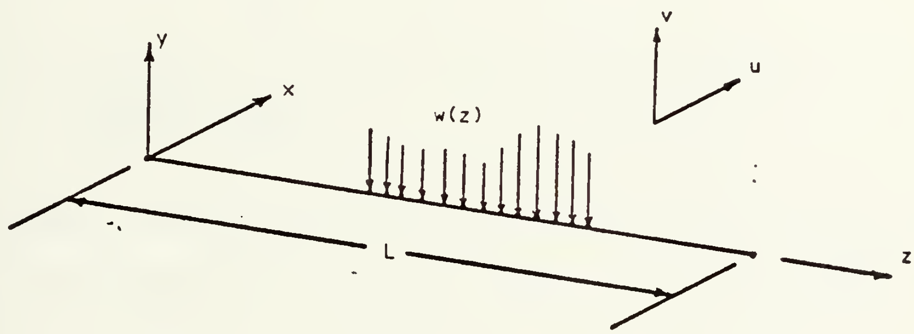

Figure 1. General configuration and coordinate system.

The beam displacements are defined by three variables: vertical deflection v, horizontal deflection u, and angular rotation ϕ. Positive senses for v and u are as indicated in figure 1. The positive sense for ϕ is indicated in figure 2. The deformations u, v and ϕ are considered to be "small" in a mathematical sense, so that, for example $\sin \phi \approx \phi$.

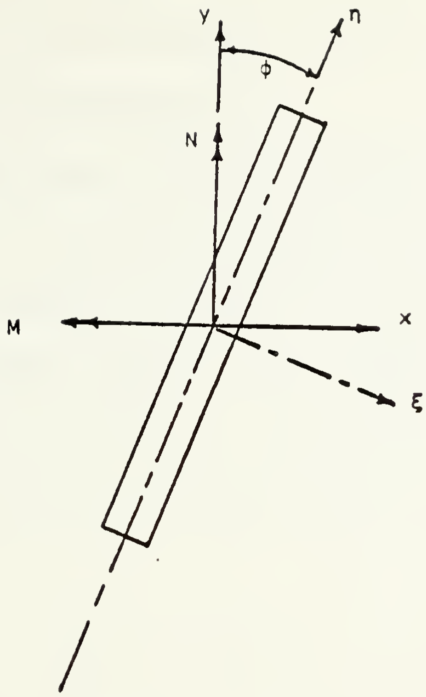

Figure 2. Positive sense of ϕ and source of bending moment $N-\phi M$.

Defining the problem requires specifications for five physical characteristics of the beam. They are beam length L, flexural rigidity for bending in the vertical plane EI1,* flexural rigidity for bending in the horizontal plane EI2, torsional rigidity C, and warping rigidity C1. The last four may vary along the length of the beam.

Torsional rigidity C is defined by the following equation.

$$C = (\text{TORQUE})/(\text{ANGLE OF TWIST PER UNIT LENGTH}) \quad (1)$$

---

*For simplicity in notation, we use numerical suffixes rather than numerical subscripts.

Warping rigidity C1 is related to nonuniform torsion. A discussion here of this property would be too digressive; cf. section 5.3 of reference (2).

It is the underlying purpose of this entire analysis to determine the total externally applied load P which will cause incipient lateral buckling. The only loading considered is distributed load in the negative y direction. The <u>type</u> of loading (uniform, triangular or whatever) is specified beforehand by a dimensionless function f(z) such that

$$w(z) = (R/L^3)\ f(z) \qquad (2)$$

where the load shape function w(z) has dimensions of force per unit length. Concentrated loads and concentrated moments can be approximated by appropriate large local variations in f(z). The function f(z) will be denoted by x5 when it appears later. The constant R has dimensions of force times distance squared. The magnitude of R is arbitrary and is chosen at the convenience of the user.

The load function has been defined. We are interested in determining the multiplying factor Q such that lateral buckling <u>does not</u> take place if the actual loading is less than Qw(z) and <u>does</u> take place if the actual loading exceeds Qw(z). The entire purpose of the program developed from this analysis is to determine this multiplier Q. In order to obtain nondimensionalized equations, it is convenient to

introduce P the total applied load for incipient buckling, defined by

$$P = Q \int_0^L w(z)\, dz \tag{3}$$

We will also employ the normalization

$$\int_0^L f(z)\, dz = L \tag{4}$$

so that $$Q = PL^2/R \tag{5}$$

For the special case where the load contributions in the negative y direction are canceled by the load contributions in the positive y direction the following alternate form of equation 3 is applied.

$$P = \int_0^L Q|w(z)|dz \tag{6}$$

The load may be applied above, on, or below the centroidal axis as specified by the load eccentricity, e(z), cf. figure 6 (page 20).

For notational and programming convenience a series of dimensionless functions x are defined. The first six are used to specify beam properties and loading. A seventh is simply a function having constant unit value.

$$x1 = EI1/R \tag{7}$$

$$x2 = EI2/R \tag{8}$$

$$x3 = C/R \tag{9}$$

$$x4 = RL^2/C1 \tag{10}$$

$$x5 = w(z)\, L^3/R = f(z) \tag{11}$$

$$x6 = e(z)/L \tag{12}$$

$$x7 = 1 \tag{13}$$

The nondimensionalizing scheme used to define the x functions is applied throughout the problem development. The dimensionless forms for vertical and horizontal deflection are

$$x10 = v/QL \tag{14}$$

and

$$\beta = u/L \tag{15}$$

where β is a dimensionless function which ultimately will be constructed from several functions.

We will use the dimensionless parameter

$$\zeta = z/L \tag{16}$$

to specify axial position. Accordingly we will think of the x functions as functions of ζ, thus $x5 = x5(\zeta)$. Differentiation with respect to z and ζ are indicated as $(\frac{dY}{dz} = Y')$ and $(\frac{dY}{d\zeta} = \dot{Y})$ respectively. The two operations are related by the equation

$$\frac{d}{dz} = \frac{1}{L}\frac{d}{d\zeta} \tag{17}$$

The developments which follow require considerable integration. Two notational devices are used to facilitate both the readability and the transcription of equations. Integration is signified by p, thus

$$px3 = \int_0^\zeta x3(\zeta)d\zeta \ .$$

Evaluation of any function at the right end of the beam, i.e., for $\zeta = 1$, is indicated by the prefix *. Several examples illustrating this notation are

$$*px1 = \int_0^1 x1(\zeta)d\zeta \tag{18}$$

$$*x5x9 = x5(1) \cdot x9(1) \tag{19}$$

$$*x2\ px8 = x2(1) \int_0^1 x8(\zeta)d\zeta \tag{20}$$

$$*px5x6px11 = \int_0^1 x5(\zeta)x6(\zeta)[\int_0^\zeta x11(\theta)d\theta]d\zeta \tag{21}$$

(In the last equation θ is a dummy variable.)

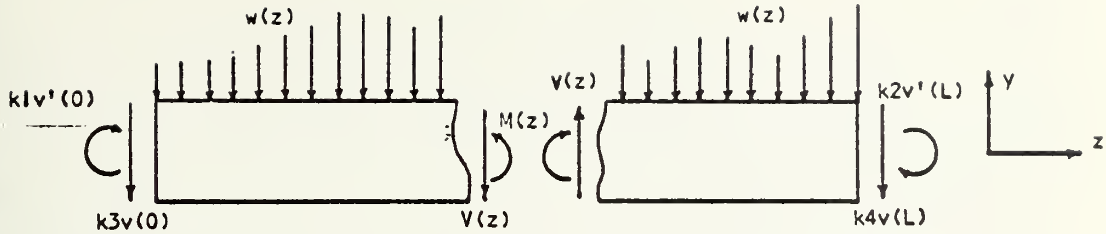

Figure 3. Diagram of forces and moments in the vertical plane.

Our analysis is restricted to beams which are unconstrained except at one or both ends. There are six possible constraints at each end each of which is modeled as a spring. The spring constant k describing such a constraint can actually vary from zero (perfectly free, no constraint at all) to infinity (perfect or ideal constraint) <u>inclusive</u>. Therefore to avoid mathematical difficulties the analysis employs dimensionless spring parameters, designated by the symbol a (with an appropriate identifying suffix). For a torsion spring, such as that corresponding to the moment k1v'(0) shown in figure 3 (page 15), the relations between k1 and a1 are

$$a1 = k1/(k1 + R/L) \tag{22}$$

$$k1 = Ra1/L(1 - a1) \tag{23}$$

Note that a1 = 0 corresponds to k1 = 0 (perfect freedom) and a1 = 1 corresponds to k1 = ∞ (perfect constraint).

Similarly, for a linear spring such as spring number 3, the relations are

$$a3 = k3/(k3 + R/L^3) \qquad (24)$$

$$k3 = Ra3/L^3(1 - a3) \qquad (25)$$

Figure 5 (page 19) shows that the point of application of a linear spring is not limited to the centroidal axis. The vertical deflection constraints may be applied at distances c1 and c2 above the centroidal axis, and the horizontal deflection constraints may be applied at distances c3 and c4 above the centroidal axis. The positive sense is as indicated in figure 5, and the dimensionless forms of these distances are

$$G1 = c1/L \qquad (26)$$

$$G2 = c2/L \qquad (27)$$

$$G3 = c3/L \qquad (28)$$

$$G4 = c4/L \qquad (29)$$

The physical and geometrical parameters used in the problem have now been defined. They will be used in the structural analysis developed in the next section.

III. STRUCTURAL ANALYSIS

There are three structural deformations examined in the analysis. They are vertical bending, horizontal bending, and torsion.

The vertical bending problem is disjoint from the remainder of the analysis, and it is solved first. The notation used in defining vertical bending will not be shown because the problem is simple and familiar. It is sufficient to say that a set of four simultaneous nonhomogeneous equations are formed, representing the constraint conditions at each end which pertain to bending in the vertical (yz) plane. This system is then solved by a standard algebraic analysis, and the result is used to create three dimensionless functions for shear, moment, and deflection.

$$\text{Shear} \quad x8 = V/P \tag{30}$$

$$\text{Moment} \quad x9 = M/PL \tag{31}$$

$$\text{Deflection} \quad x10 = v/QL \tag{32}$$

The remainder of the analysis involves studying the combined horizontal bending and torsion problem. Figure 4 (page 18) shows the positive sense for the constant horizontal shear H, the linearly varying horizontal bending moment N, and horizontal displacement u.

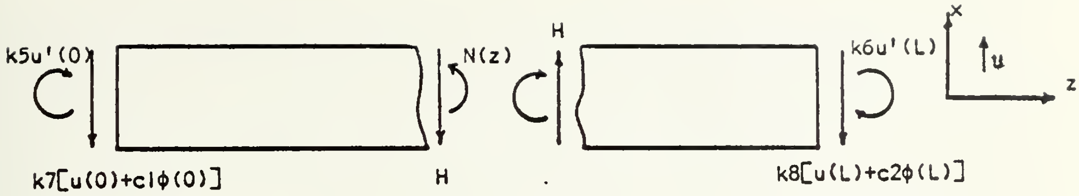

Figure 4. Diagram showing forces and moments in the horizontal plane.

The appropriate dimensionless variables are

$$\beta = u/L \; ; \quad u = L\beta \tag{33}$$

$$B4 = H/P \; ; \quad H = PB4 = B4P \tag{34}$$

$$n = N/PL ; \quad N = PLn \tag{35}$$

Dimensionless shear B4 is one of ultimately eight unknown scalar quantities (B1, B2,...., B8) which will be introduced in the process of determining Q. The alternate forms indicated in the preceding equations are intended to familiarize the reader with notational forms which will be freely employed in later developments.

The illustration of the rotated cross section in figure 2 (page 10) shows that the moment causing bending about the η axis is N cos ϕ - M sin ϕ, which reduces to N - Mϕ by small angle approximation. From elementary bending theory we have

$$EI2\, u'' = N - \phi M \tag{36}$$

and by substitution from equation 7 it may be written as

$$\ddot{\beta} = Qx2(n - \phi x9) \tag{37}$$

Examination of the simple statics of figure 4 (page 18) yields

$$Hz + N(0) = N(z) \tag{38}$$

In dimensionless form this becomes

$$B4\zeta + n(0) = n(\zeta) \tag{39}$$

and may be rewritten as

$$n = B4px7 + B5x7 \tag{40}$$

$$n(0) = B5 \tag{41}$$

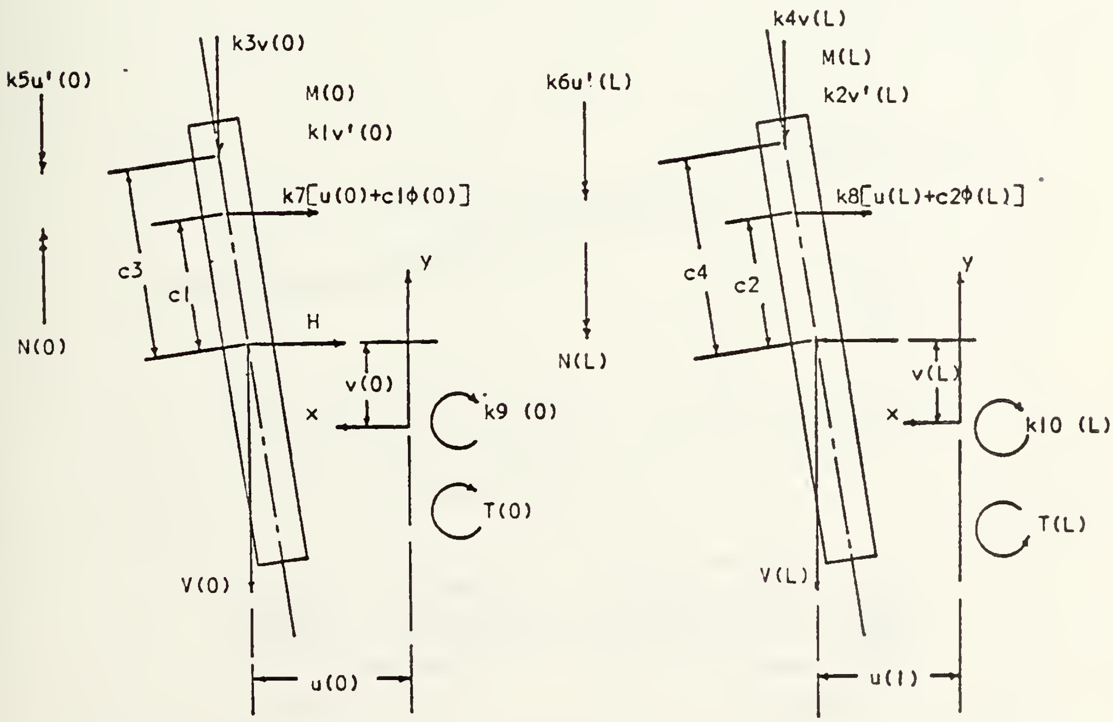

Figure 5. Force and moment diagrams of end sections.

Figures 5 and 6 (page 20) show the forces and moments used to develop the torsion moment equation

$$T(z) = T(0)-H[v(z)-v(0)] + V(z)u(z)-V(0)u(0)$$

$$+ \int_{0}^{z} w(z)[u(z)+e(z)\phi(z)]dz \qquad (42)$$

Introducing the dimensionless function

$$t = T/PL \qquad (43)$$

and using additional dimensionless functions previously defined, equation 31 may be written as

$$t = t(0)-QB4[x10-x10(0)]+x8\beta-x8(0)\beta(0)+px5(\beta+x6\phi) \qquad (44)$$

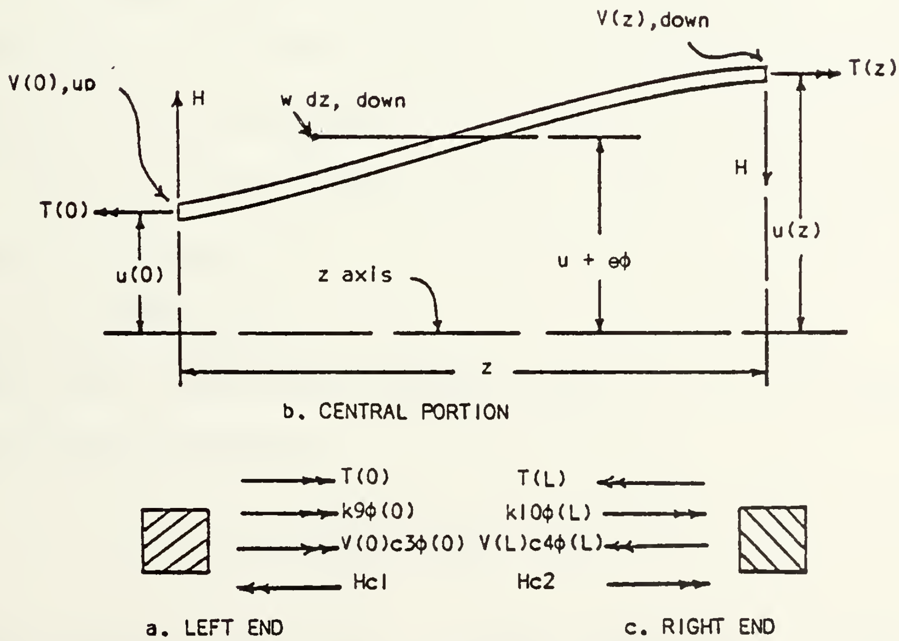

Figure 6. Diagram of left, central, and right portions for torsion analysis.

Reference (1) develops the following third order equation which accounts for the warping rigidity of the beam.

$$C1\phi''' = C\phi' + T - u'M \tag{45}$$

In dimensionless form this may be written as

$$\dddot{\phi} = x4(Qt - Qx9\dot{\beta} + x3\dot{\phi}) \tag{46}$$

The function x4 has previously been defined as

$$x4 = RL^2/C1 \tag{47}$$

C1 is the coefficient of the highest (i.e., third) order derivative in equation 45, and if C1 becomes small, very troublesome mathematical difficulties are encountered, cf. Reference (3). Accordingly, the analysis for the case C1 $\equiv 0$ cannot be obtained from that for C1 $\neq 0$, given in reference (1), except with the greatest of difficulty. It is for this reason that the separate analysis represented by this thesis, was undertaken. The analysis herein, to this point, is, as has been said before, essentially identical to that in reference (1). From this point on there are essential differences.

The present analysis presumes that C1 = 0, so that equation 35 takes the form

$$\dot{\phi} = Qx3^{-1}(x9\dot{\beta} - t) \tag{48}$$

The inverse of x3, $x3^{-1}$, occurs frequently in what follows. The term is clumsy in this form and will be redesignated as

$$Y = x3^{-1} \tag{49}$$

Therefore equation 48 is written as

$$\dot{\phi} = QY(x9\dot{\beta} - t) \tag{50}$$

Two of the end constraint conditions of the original problem, namely those relating to warping constraints at the end, are not available for and do not appear in the present specialized problem.

The original problem had six constraints at each end. The present problem has five constraints at each end. Four constraint conditions were employed in solving the problem of bending in the vertical plane. Thus a total of six constraint conditions remain at this stage of the development. The first four result from the constraints against horizontal deflection and end rotations in the x-z plane. The last two result from constraints against rotation about the z axis.

$$a5\dot{\beta}(0) - Q(1-a5)\, n(0) = 0 \tag{51}$$

$$a6\dot{\beta}(1) + Q(1-a6)\, n(1) = 0 \tag{52}$$

$$a7[\beta(0)+G1\phi(0)] + QB4(1-a7) = 0 \tag{53}$$

$$a8[\beta(1)+G2\phi(1)] - QB4(1-a8) = 0 \tag{54}$$

$$a9\phi(0)+Q(1-a9)[t(0)+G3x8(0)\phi(0)-B4G1] = 0 \tag{55}$$

$$a10\phi(1)-Q(1-a10)[t(1)+G4x8(1)\phi(1)-B4G2] = 0 \tag{56}$$

Although this notation appears formidable, it is convenient, explicit, and unequivocal. The terms are

sufficiently complicated that no matter what notation might be used careful attention to detail is required.

There is no need to demonstrate the complete development of the six boundary equations, but for illustrative purposes the derivations of two typical constraint equations, 51 and 53, will be given.

The constraint against rotation of the left end about a vertical (y) axis, as shown in figure 6 (page 20) results in the following equation.

$$k5u'(0) - N(0) = 0 \qquad (57)$$

Employing the dimensionless variable $\beta = u/L$ and $n = N/PL$, using equation 17, writing the spring constant k5 in terms of the spring parameter a5, viz,

$$k5 = Ra5/L(1-a5) \qquad (58)$$

and recalling equation 5, it is easy to obtain the equation

$$a5\dot{\beta}(0)-Q(1-a5)n(0) = 0 \qquad (59)$$

This is equation 51.

Equation 53 defines the constraint against horizontal deflection of the left end of the beam. The deflection is $u(0) + c1\phi(0)$, see figure 5 (page 19), where the second term results from permitting the spring to be attached at a distance c1 above the centroidal axis. The constraint equation is

$$k7[u(0) + c1\phi(0)] + H = 0 \qquad (60)$$

where as previously defined H = PB4, c1 = LG1, and

$$k7 = a7R/L^3(1-a7) \tag{61}$$

Using the dimensionless quantities previously introduced, it is a simple matter to put equation 60 in the form exhibited as equation 53.

The structural analysis has now been completed and is represented by equations 37, 44, 50, 51, 52, 53, 54, 55, and 56. The mathematical method of solution will be addressed next.

IV. MATHEMATICAL ANALYSIS

Before going into the details of the analysis, we will try to describe it in general terms.

An expression is formed for $\dot{\phi}$ which involves an unknown function (x11 in what follows) the initial form of which is arbitrarily assumed except for the normalizations

$$x11(0) = 0 \tag{62}$$

$$*p|x11| = 1 \tag{63}$$

The corresponding function ϕ is obtained by integration and equation 37 is used to get $\ddot{\beta}$. This is followed by two more integrations to get $\dot{\beta}$ and β. These are employed in the torsion equation 44 to get the function t. Finally this is used in equation 48 to get a new expression for $\dot{\phi}$. Each integration introduces a new unknown scalar quantity. In one way or another a total of eight such unknown scalar constants B1, B2,...,B8 are introduced. The six constraint equations, plus two more which will be introduced in what follows, provide a set of eight linear homogeneous equations in these unknown B's which thus leads to an eigenproblem of eighth order. The parameter Q appears in the coefficients to the first and second powers. The quadratic eighth order eigenproblem is transformed to a linear sixteenth order eigenproblem. Since, generally, both zero and infinite eigenvalues are contained in the solution, a special, relatively new

algorithm, the QZ method, capable of dealing with such systems, is employed for the solution of this eigenproblem. The appropriate eigenvalue is chosen and the corresponding eigenvector provides the solution for the unknown B's so that the new x11 can be uniquely determined. This new x11 is used in place of the original x11 and the process is repeated until there is convergence. No proof is offered that indeed convergence must ultimately take place; because of the complexity of the problem such a proof might be very difficult to establish. However, experience with the procedure described here indicates that convergence may indeed be expected. The details of this procedure are now given in what follows.

The function $\dot{\phi}$ is initially specified as

$$\dot{\phi} = B1x11 + B2x7 \tag{64}$$

where x11 is an arbitrarily assumed function satisfying equations 62 and 63 and where B1 and B2 are the third and fourth of eight unknown constants that will be employed. The first was B4 from equation 34 and the second was B5 from equation 41.

Integrating equation 64 yields

$$\phi = B1px11 + B2px7 + B3x7 \tag{65}$$

and when equations 40 and 65 are substituted into equation 37 we arrive at

$$\ddot{\beta} = -B1Qx2x9px11 - B2Qx2x9px7 - B3Qx2x9$$
$$+B4Qx2px7 + B5Qx2 \tag{66}$$

Integrating twice more, we get

$$\dot{\beta} = -B1Qpx2x9px11 - B2Qpx2x9px7 - B3Qpx2x9$$
$$+B4Qpx2px7 + B5Qpx2 + B6Qx7 \tag{67}$$

$$\beta = -B1Qppx2x9px11 - B2Qppx2x9px7 - B3Qppx2x9$$
$$+B4Qppx2px7 + B5Qppx2 + B6Qpx7 + B7Qx7 \tag{68}$$

For convenience the next step is to rewrite equation 44 as

$$t = q\beta + px5x6\phi - QB4X - Qx8(0)B7 + QB8 \tag{69}$$

where
$$t(0) = QB8 \tag{70}$$
$$X = x10 - x10(0) \tag{71}$$
$$q = px5 + x8 \tag{72}$$

The symbol q denotes an operator acting on the function β, not a function which is multiplied by β.

When the preceding equations, evaluated at $\zeta=0$ or $\zeta=1$ as the case may be, are substituted in boundary conditions equations 51 through 56, a system of six linear simultaneous homogeneous equations in eight unknown constants B is obtained.

In the previous section, which addressed the structural analysis, equations 51 and 53 were derived for illustrative purposes. Now for illustrative continuity these two equations will be used to show the method of formulation of the six linear simultaneous homogeneous equations.

Recalling that equation 51 is

$$a5\dot{\beta}(0) - Q(1-a5)\, n(0) = 0$$

and substituting

$$\dot{\beta}(0) = QB6 \tag{73}$$

$$n(0) = B5 \tag{74}$$

we now factor out Q leaving

$$-B5(1-a5) + B6a5 = 0 \tag{75}$$

The second illustrative equation, 53, is

$$a7[\beta(0) + G1\phi(0)] + QB4(1-a7) = 0$$

and upon substituting

$$\phi(0) = B3 \tag{76}$$

$$\beta(0) = B7Q \tag{77}$$

we arrive at

$$B3G1a7 + B4Q(1-a7) + B7Qa7 = 0 \tag{78}$$

If equations 67 and 69 are substituted into equation 50 the following "new" $\dot{\phi}$ is created.

$$
\begin{aligned}
\dot{\phi} = Y[&B1(-Q^2 x9px2x9px11 + Q^2 qppx2x9px11 - Qpx5x6px11) \\
&+ B2(-Q^2 x9px2x9px7 + Q^2 qppx2x9px7 - Qpx5x6px7) \\
&+ B3(-Q^2 x9px2x9 + Q^2 qppx2x9 - Qpx5x6) \\
&+ B4(Q^2 x9px2px7 + Q^2 X - Q^2 qppx2px7) \\
&+ B5(Q^2 x9px2 - Q^2 qppx2) \\
&+ B6(Q^2 x9 - Q^2 qpx7) \\
&+ B8(-Q^2)]
\end{aligned} \tag{79}
$$

(The coefficient of B7 vanishes identically.)

To complete the eigensystem involving the eight unknown B's, two additional equations are needed.

The seventh equation, consistent with equation 62, is

$$\text{new } \dot{\phi}(0) - B2 = 0 \tag{80}$$

The eighth equation is

$$(*p\dot{\phi})_{old} - (*p\dot{\phi})_{new} = 0 \tag{81}$$

and is arrived at by observing that if $\dot{\phi}$ given in equation 64 and created from an assumed x11, were, somehow, a "correct" $\dot{\phi}$ then it and the $\dot{\phi}$ recovered in equation 79 would be identically equal to each other. The integration in equation 81 achieves a certain "smoothing" or averaging which has been found to be preferable to simply equating the new and old functions at an arbitrary value of ζ.

The coefficients of the eigensystem equations generally involve Q to the zeroth, first, and second powers so that the system may be exhibited in the form

$$(D + EQ + FQ^2)B = 0 \tag{82}$$

The nonzero elements of matrices D, E, and F are

D15 = a5-1, D16 = a5, D21 = -a6*px2x9px11,

D22 = -a6*px2x9px7, D23 = -a6*px2x9, D24 = a6*px2px7-a6+1,

D25 = a6*px2-a6+1, D26 = a6, D33 = G1a7,

D41 = a8G2*px11, D42 = a8G2, D43 = a8G2,

D53 = a9, D61 = a10*px11, D62 = a10, D63 = a10,

D72 = -1, D81 = -*px11, D82 = -1, E34 = 1-a7,

E37 = a7, E41 = -a8*ppx2x9px11,

E42 = -a8*ppx2x9px7, E43 = -a8*ppx2x9,

E44 = a8*ppx2px7+a8-1, E45 = a8*ppx2,

E46 = a8, E47 = a8, E53 = (1-a9)G3x8(0),

E54 = (a9-1)G1, E61 = (a10-1)[*px5x6px11+G4*x8(1)px11],

E62 = (a10-1)[*px5x6px7 + G4*x8(1)],

E63 = (a10-1)[*px5x6+G4*x8(1)], E64 = (1-a10)G2,

E81 = -*pYpx5x6px11, E82 = -*pYpx5x6px7,

E83 = -*pYpx5x6, F58 = 1-a9,

F61 = (1-a10)*qppx2x9px11, F62 = (1-a10)*qppx2x9px7,

F63 = (1-a10)*qppx2x9, F64 = (1-a10)(*X-*qppx2px7),

F65 = (a10-1)*qppx2, F66 = (a10-1)*qpx7,

F68 = a10-1, F76 = x9(0)Y(0), F78 = -Y(0),

F81 = -*pYx9px2x9px11 + *pYqppx2x9px11,

F82 = -*pYx9px2x9px7 + *pYqppx2x9px7,

F83 = -*pYx9px2x9 + *pYqppx2x9,

F84 = *pYx9px2px7 + *pYX - *pYqppx2px7,

F85 = *pYx9px2 - *pYqppx2,

F86 = *pYx9 - *pYqpx7,

F88 = -*pY

[For reference purposes all the preceding equations bear the number 84.]

This eighth order quadratic eigensystem may be expanded to a sixteenth order linear eigensystem in the form

$$VZ = QWZ \tag{85}$$

where

$$V = \begin{bmatrix} E & D \\ I & O \end{bmatrix} \tag{86}$$

$$W = \begin{bmatrix} -F & O \\ O & I \end{bmatrix} \tag{87}$$

$$Z = \begin{bmatrix} QB \\ B \end{bmatrix} \tag{88}$$

and I and O respectively represent the eighth order unit and null matrices.

The eigenproblem has now been defined in a linear form, but because the matrices V and W may be singular not just any method will satisfactorily solve the problem. We have chosen the QZ method described in reference (3). From the sixteen eigenvalues produced the smallest positive eigenvalue is selected. The last eight elements of the associated eigenvector are the desired constants B1 through B8.

Using equation 79 a new $\dot{\phi}$ is now formed and from it a new xll is created by means of the following normalization process.

$$xll_{new} = [\dot{\phi}_{new} - \dot{\phi}_{new}(0)]/[*p|\dot{\phi}_{new} - \dot{\phi}_{new}(0)|] \tag{89}$$

The iteration process is now repeated until satisfactory convergence is attained. The converged smallest positive eigenvalue Q is now used to determine the buckling load from either equation 3 or equation 6.

V. COMPUTER IMPLEMENTATION OF THEORY

To this point the functions x have been treated as continuous functions for use in an analytic solution. However, it is obvious that solving the problem, even for simple functions, by analytic means is prohibitively difficult. To overcome this problem a numerical method of solution was employed. The beam was divided into a finite number of equal sections. The functions x were then redefined as vectors with Fortran designations x(1,I), x(2,I)..., etc. where I = 1, 2, 3,...,N. The elements of any such vector represent function values at the N points of subdivision of the beam into equal subsections of length L/(N-1). The vector x(10,I) was used twice, first to represent x10 and then to store X of equation 71.

The vectors in this form readily lend themselves to manipulation. The simple subroutines developed to perform the operations are, with the exception of the integration scheme, not worthy of note here.

Trapezoidal integration was first used during the debugging of the raw, untested program. This method gives satisfactory results with an error of $O(h^2)$, where h is the interval length. Once it was established that the program would give satisfactory solutions the refinement process was begun. A higher order numerical integration scheme was employed next. It is described by Milne in reference (5),

and results in an error $O(h^5)$.

The QZ subroutine is a canned program in the library of the Naval Postgraduate School Computer Center. It was transcribed from reference (6), with very little change, by Mr. Roger Hilleary. To date, it has been the only program that has given satisfactory solutions to the present eigenvalue problem.

The complete theory described in the preceding sections of this thesis has been implemented in the form of a subroutine LATBRO, a complete listing of which is contained in Appendix A.

The user must write a main program which supplies dimensioning statements, number of intervals, specifications for vectors x1 through x6, specifications for the constants G, and the ten spring parameters. The following is an example of such a main program. It was used to solve what is listed as Case 6, Table 1, page 37.

```
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 X(30,101),ALFA(12),G(4),S,ONE,ZERO
      INTEGER KP(10)
      COMMON X,ALFA,G,N,KP
      N=101
      ZERO=0.D+0
      ONE=1.D+0
      DO 1 I=1,N
      X(1,I)=ONE
      X(2,I)=ONE
      X(3,I)=ONE
      X(5,I)=ZERO
      X(6,I)=ZERO
    1 CONTINUE
      X(5,1)=ONE
      DO 2 I=1,4
    2 G(I)=ZERO
      DO 3 I=1,12
```

```
3 ALFA(I)=ZERO
  ALFA(2)=ONE
  ALFA(4)=ONE
  ALFA(6)=ONE
  ALFA(8)=ONE
  ALFA(10)=ONE
  KP(3)=1
  CALL LATBRO(S)
  STOP
  END
```

VI. TESTING AND VERIFICATION

As mentioned in the previous section, the numerical integration methods that have been employed have their shortcomings.

If the integrand is "smooth," either the trapezoidal or the higher order Milne method may be used. The latter is preferable because of its greater accuracy.

Both methods introduce errors, however, when there are steps, impulses, or doublets to be integrated. We have found that the Milne and trapezoidal methods can be used to solve problems with a concentrated load. The error depends linearly on the reciprocal of the number of elements into which the beam is divided and extrapolation may be used to obtain an excellent result. As yet we do not have experience with concentrated moment loadings (doublets). This subject is discussed in more detail in reference (1).

The following table shows the cases that have been tested to date. In each of these cases all the G's and the eccentricity x6 were zero. Each of these has been successful.

The values for Q shown in the last column of the table were obtained with $x2 \equiv 1$, $x3 \equiv 1$, and by simple scaling analysis lead to total load

$$P = Q \sqrt{CEI2}/L^2 \qquad (89)$$

TABLE I

Cases of Uniform Beams Analyzed by Subroutine LATBRO
($x6 = 0$, $G1 = G2 = G3 = G4 = 0$)

Case Number	Values of Spring Parameters a1	a2	a3	a4	a5	a6	a7	a8	a9	a10	Loading Type	Q
1	0	1	0	1	0	1	0	1	0	1	UNIF	12.854
2	1	0	1	0	1	0	1	0	1	0	UNIF	12.854
3	0	0	1	1	1	1	1	1	1	1	UNIF	47.58
4	1	1	1	1	0	0	1	1	1	1	UNIF	97.11
5	0	0	1	1	0	0	1	1	1	1	CONC MIDSPAN	16.936
6	0	1	0	1	0	1	0	1	0	1	CONC LEFT END	4.018

for lateral buckling. The values reported agree with values to be found in reference (2).

The following are details of several cases.

Case 1

As shown, it is a cantilever beam fully restrained at the right end. The loading is uniform.

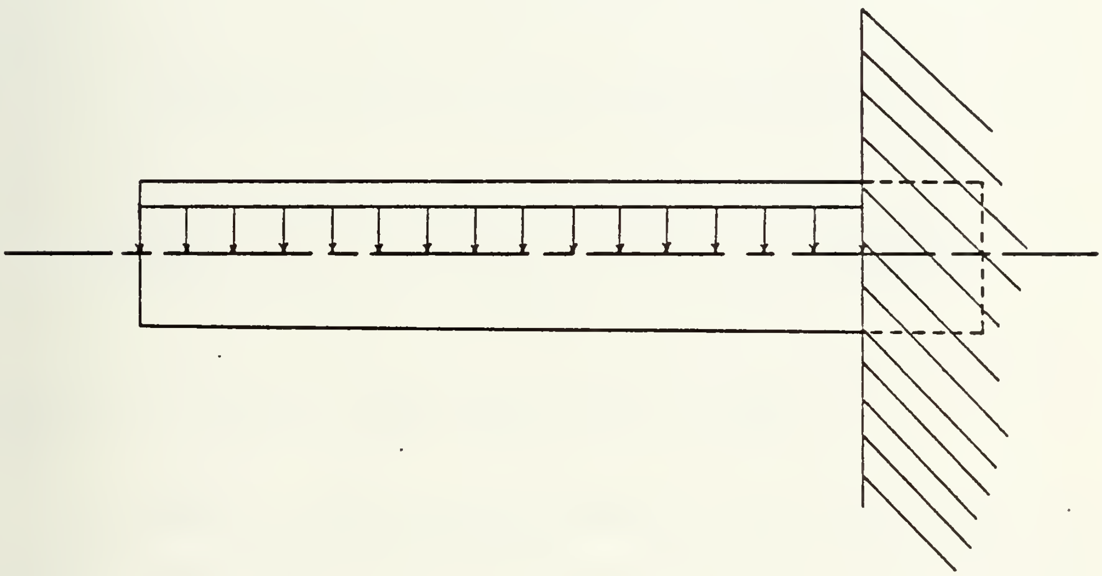

Figure 7. Illustration of side view of Case 1 beam.

The subroutine produced a converged value of Q of 12.854. Timoshenko's result is 12.85; cf. page 261 ref (2).

Case 2

This is simply Case 1 turned end for end. It was used to test additional equations and also resulted in a converged critical load output of 12.854 which is identical to the result for Case 1, as it should be.

Case 5

The converged critical load output was 16.936 and Timoshenko's result is 16.94; cf. page 269 ref (2).

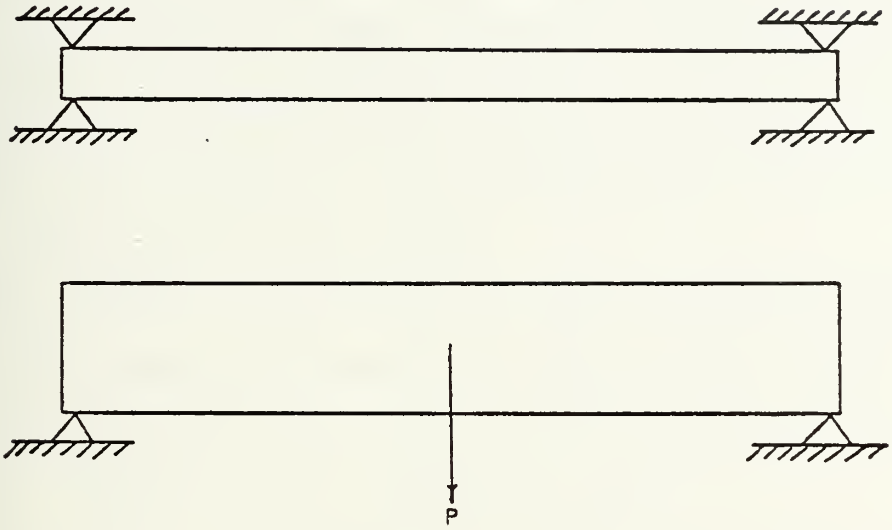

Figure 8. Top and side views Case 5 beam.

In several of the cases listed in the table above, the loading does not interact with one or more of the springs, so that the corresponding spring parameters could have an arbitrary value without changing the results.

VII. CONCLUSIONS AND RECOMMENDATIONS

The subroutine LATBRO in its present form should be able to solve beam problems with beams that have uniform or variable cross sections, with loading on the centroidal axis, and with spring constraints attached at the centroidal axis. Loading that involves concentrated moments, as mentioned before, has not yet been tested.

The program was designed to solve problems where loading and spring constraints may act away from the centroidal axis. It is recommended that follow on testing begin with

1. loading off the centroidal axis.
2. spring constraints attached away from the centroidal axis.
3. concentrated moment loading.

It is expected, with the exception of minor programming errors, that problems with any combination of the above will be readily solved by the subroutine.

APPENDIX A

Listing of LATBRO and Adjunct Subroutines

```
      SUBROUTINE LATBRO(S)
      IMPLICIT REAL*8(A-H,O-Z)
      REAL*8 X(30,101),ALFA(12),G(4),S
      REAL*8 WKAREA(800),ZERO,ONE,ONEG,DELTA,LAM
      REAL*8 C(4,4),CC(4),D(8,8),E(8,8),F(8,8),B(8),DEN(16),EN
      REAL*8 AA(16,16),BB(16,16)
      REAL*8 EKS(16,16),NUMR(16),NUMI(16)
      REAL*8 RESULT(50)
      INTEGER ITER(16)
      INTEGER KP(10)
      LOGICAL WANTX
      COMMON X,ALFA,G,N,KP
      KKK=1
      ZERO=0.D+0
      ONE=1.D+0
      ONEG=-ONE
      DO 1 I=1,N
    1 X(7,I)=ONE
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C  THE FOLLOWING PRINTS ADVICE TO THE USER THAT HE HAS CHOSEN THE
C FIRST ORDER CASE WHICH MAKES NO CONSIDERATION OF WARPING.
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      WRITE(6,2004)
 2004 FORMAT('0',' USER NOTE: YOU HAVE CHOSEN THE USE OF THE FIRST ORDER
     C SUBROUTINE AND NO CONSIDERATION WILL BE GIVEN TO THE WARPING PHEN
     COMENON.')
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C SOLVE THE X,Z BENDING PROBLEM
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C  THE FIRST STEP WILL BE TO NORMALIZE THE LOADING VECTOR X5.
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      CALL NORV(5)
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C NOW PREPARE THE COEFFICIENT ARRAYS FOR THE X,Z BENDING PROBLEM.
```

```
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      DO 2 I=1,4
      CC(I)=ZERO
      DO 2 J=1,4
    2 C(I,J)=ZERO
      C(1,2)=ALFA(1)-ONE
      C(1,3)=ALFA(1)
      C(2,3)=ALFA(2)
      C(3,1)=ONE-ALFA(3)
      C(3,4)=ALFA(3)
      C(4,4)=ALFA(4)
      CALL INTV(5,13)
      CC(4)=(ALFA(4)-ONE)*X(13,N)
      CALL INTV(13,12)
      CC(2)=(ONE-ALFA(2))*X(12,N)
      CALL MULV(12,1,16)
      CALL INTV(16,11)
      CC(2)=CC(2)+ALFA(2)*X(11,N)
      CALL INTV(11,16)
      CC(4)=CC(4)+ALFA(4)*X(16,N)
      CALL INTV(7,14)
      C(4,3)=ALFA(4)*X(14,N)
      C(2,1)=(ONE-ALFA(2))*X(14,N)
      CALL MULV(1,14,16)
      CALL INTV(16,11)
      C(2,1)=C(2,1)+ALFA(2)*X(11,N)
      CALL INTV(11,16)
      C(4,1)=ALFA(4)*X(16,N)+ALFA(4)-ONE
      CALL INTV(1,16)
      C(2,2)=ALFA(2)*X(16,N)+ONE-ALFA(2)
      CALL INTV(16,11)
      C(4,2)=ALFA(4)*X(11,N)
      DO 1000 I=1,4
 1000 WRITE(6,1002)(C(I,J),J=1,4),CC(I)
 1002 FORMAT(1X,1P5E20.8)
CT    DO 1000 I=1,4
C1000 WRITE(6,1002)(C(I,J),J=1,4),CC(I)
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C NOW USING A CANNED SUBROUTINE TO SOLVE THE LINEAR SYSTEM.
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      M=1
      NA=4
```

```
      IDGT=0
      CALL LEQT1F(C,M,NA,NA,CC,IDGT,WKAREA,IER)
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C  VECTOR CC NOW CONTAINS (DIMENSIONLESS) SHEAR, MOMENT,
C  SLOPE, AND DEFLECTION AT THE LEFT END.
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      CALL SUBS(13,20,CC(1))
      CALL MULS(20,8,ONEG)
      CALL INTV(8,9)
      CALL ADDS(9,9,CC(2))
      CALL MULV(1,9,10)
      CALL INTV(10,11)
      CALL ADDS(11,11,CC(3))
      CALL INTV(11,10)
      CALL ADDS(10,10,CC(4))
C     CALL PRIV(8,2,8)
      CALL PRIV(8,2,8)
C     CALL PRIV(9,2,9)
      CALL PRIV(9,2,9)
C     CALL PRIV(10,2,10)
      CALL PRIV(10,2,10)
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C  VECTORS 8 (SHEAR), 9 (MOMENT), AND 10 (DEFLECTION) HAVE BEEN
C  DETERMINED.  NOW WE ARE READY TO SOLVE THE LATERAL BUCKLING
C PROBLEM. FIRST WE ZERO D(8,8),E(8,8), AND F(8,8).
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      DO 10 I=1,8
      DO 10 J=1,8
      D(I,J)=ZERO
      E(I,J)=ZERO
   10 F(I,J)=ZERO
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C  NOW WE CALCULATE THOSE COEFFICIENTS WHICH DO NCT DEPEND ON X11.
C  THE FIRST COMMAND BELOW TURNS X10 UNTO CAPITAL X10.
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      CALL SUBS(10,10,X(10,1))
```

```
      D(1,5)=ALFA(5)-ONE                                                BRK11720
      D(1,6)=ALFA(5)                                                    BRK11730
      D(2,6)=ALFA(6)                                                    BRK11740
      D(3,3)=ALFA(7)*G(1)                                               BRK11750
      E(3,4)=ONE-ALFA(7)                                                BRK11760
      E(3,7)=ALFA(7)                                                    BRK11770
      D(4,3)=ALFA(8)*G(2)                                               BRK11780
      D(4,2)=ALFA(8)*G(2)                                               BRK11790
      E(4,6)=ALFA(8)                                                    BRK11800
      E(4,7)=ALFA(8)                                                    BRK11810
      E(5,3)=(ONE-ALFA(9))*G(3)*X(8,1)                                  BRK11820
      D(5,3)=ALFA(9)                                                    BRK11830
      E(5,4)=(ALFA(9)-ONE)*G(1)                                         BRK11840
      F(5,8)=ONE-ALFA(9)                                                BRK11850
      D(6,2)=ALFA(10)                                                   BRK11860
      D(6,3)=ALFA(10)                                                   BRK11870
      E(6,4)=(ONE-ALFA(10))*G(2)                                        BRK11880
      F(6,8)=ALFA(10)-ONE                                               BRK11890
      D(7,2)=ONEG                                                       BRK11900
      D(8,2)=ONEG                                                       BRK11910
      CALL DIVV(7,3,14)                                                 BRK11920
      F(7,8)=-X(14,1)                                                   BRK11930
      F(7,6)=X(14,1)*X(9,1)                                             BRK11940
      CALL INTV(14,15)                                                  BRK11950
      F(8,8)=-X(15,N)                                                   BRK11960
      CALL INTV(7,12)                                                   BRK11970
      CALL MULV(9,12,13)                                                BRK11980
      CALL MULV(2,13,20)                                                BRK11990
      CALL INTV(20,19)                                                  BRK12000
      D(2,2)=-ALFA(6)*X(19,N)                                           BRK12010
      CALL INTV(19,17)                                                  BRK12020
      CALL QVEC(17,18)                                                  BRK12030
      E(4,2)=-ALFA(8)*X(17,N)                                           BRK12040
      F(6,2)=(ONE-ALFA(10))*X(18,N)                                     BRK12050
      CALL MULV(2,9,14)                                                 BRK12060
      CALL INTV(14,15)                                                  BRK12070
      D(2,3)=-ALFA(6)*X(15,N)                                           BRK12080
      CALL INTV(15,16)                                                  BRK12090
      E(4,3)=-ALFA(8)*X(16,N)                                           BRK12100
      CALL QVEC(16,17)                                                  BRK12110
      F(6,3)=(ONE-ALFA(10))*X(17,N)                                     BRK12120
      CALL MULV(2,12,14)                                                BRK12130
      CALL INTV(14,15)                                                  BRK12140
      D(2,4)=ALFA(6)*X(15,N)-ALFA(6)+ONE                                BRK12150
      CALL INTV(15,16)                                                  BRK12160
      E(4,4)=ALFA(8)*X(16,N)+ALFA(8)-ONE                                BRK12170
      CALL INTV(2,14)                                                   BRK12180
      D(2,5)=ALFA(6)*X(14,N)-ALFA(6)+ONE                                BRK12190
```

```
      CALL INTV(14,15)                                                  BRK12200
      E(4,5)=ALFA(8)*X(15,N)                                            BRK12210
      CALL QVEC(15,14)                                                  BRK12220
      F(6,5)=(ALFA(10)-ONE)*X(14,N)                                     BRK12230
      CALL MULV(6,12,13)                                                BRK12240
      CALL MULV(5,13,14)                                                BRK12250
      CALL INTV(14,15)                                                  BRK12260
      E(6,2)=((ALFA(10)-ONE)*X(15,N))+((ALFA(10)-CNE)*G(4)*X(8,N))      BRK12270
      CALL MULV(5,6,13)                                                 BRK12280
      CALL INTV(13,14)                                                  BRK12290
      E(6,3)=((ALFA(10)-ONE)*X(14,N))+((ALFA(10)-ONE)*G(4)*X(8,N))      BRK12300
      CALL MULV(2,12,13)                                                BRK12310
      CALL INTV(13,14)                                                  BRK12320
      CALL INTV(14,15)                                                  BRK12330
      CALL QVEC(15,16)                                                  BRK12340
      F(6,4)=(ONE-ALFA(10))*(X(10,N)-X(16,N))                           BRK12350
      CALL QVEC(12,13)                                                  BRK12360
      F(6,6)=(ALFA(10)-ONE)*X(13,N)                                     BRK12370
      CALL QVEC(7,13)                                                   BRK12380
      S=X(8,1)                                                          BRK12390
      CALL SUBS(13,13,S)                                                BRK12400
      F(6,7)=(ALFA(10)-ONE)*X(13,N)                                     BRK12410
C                                                                       BRK12420
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * BRK12430
C                                                                       BRK12440
C NOW WORKING ON TERMS DIVIDED BY X3 BUT NOT INVOLVING X11.             BRK12450
C                                                                       BRK12460
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * BRK12470
C                                                                       BRK12480
      CALL MULV(6,12,13)                                                BRK12490
      CALL MULV(5,13,14)                                                BRK12500
      CALL INTV(14,15)                                                  BRK12510
      CALL DIVV(15,3,16)                                                BRK12520
      CALL INTV(16,17)                                                  BRK12530
      E(8,2)=-X(17,N)                                                   BRK12540
      CALL MULV(9,12,13)                                                BRK12550
      CALL MULV(2,13,14)                                                BRK12560
      CALL INTV(14,15)                                                  BRK12570
      CALL INTV(15,16)                                                  BRK12580
      CALL QVEC(16,17)                                                  BRK12590
      CALL MULV(9,15,18)                                                BRK12600
      CALL SUBV(17,18,19)                                               BRK12610
      CALL DIVV(19,3,20)                                                BRK12620
      CALL INTV(20,13)                                                  BRK12630
      F(8,2)=X(13,N)                                                    BRK12640
      CALL MULV(5,6,13)                                                 BRK12650
      CALL INTV(13,14)                                                  BRK12660
      CALL DIVV(14,3,15)                                                BRK12670
```

```
      CALL INTV(15,16)                                                  BRK12680
      E(8,3)=-X(16,N)                                                   BRK12690
      CALL MULV(2,9,13)                                                 BRK12700
      CALL INTV(13,14)                                                  BRK12710
      CALL INTV(14,15)                                                  BRK12720
      CALL QVEC(15,16)                                                  BRK12730
      CALL MULV(9,14,17)                                                BRK12740
      CALL SUBV(16,17,18)                                               BRK12750
      CALL DIVV(18,3,19)                                                BRK12760
      CALL INTV(19,20)                                                  BRK12770
      F(8,3)=X(20,N)                                                    BRK12780
      CALL MULV(2,12,13)                                                BRK12790
      CALL INTV(13,14)                                                  BRK12800
      CALL MULV(9,14,15)                                                BRK12810
      CALL INTV(14,16)                                                  BRK12820
      CALL QVEC(16,17)                                                  BRK12830
      CALL ADDV(15,10,13)                                               BRK12840
      CALL SUBV(13,17,14)                                               BRK12850
      CALL DIVV(14,3,16)                                                BRK12860
      CALL INTV(16,18)                                                  BRK12870
      F(8,4)=X(18,N)                                                    BRK12880
      CALL INTV(2,13)                                                   BRK12890
      CALL INTV(13,14)                                                  BRK12900
      CALL MULV(9,13,15)                                                BRK12910
      CALL QVEC(14,16)                                                  BRK12920
      CALL SUBV(15,16,17)                                               BRK12930
      CALL DIVV(17,3,18)                                                BRK12940
      CALL INTV(18,19)                                                  BRK12950
      F(8,5)=X(19,N)                                                    BRK12960
      CALL QVEC(12,13)                                                  BRK12970
      CALL SUBV(9,13,14)                                                BRK12980
      CALL DIVV(14,3,15)                                                BRK12990
      CALL INTV(15,16)                                                  BRK13000
      F(8,6)=X(16,N)                                                    BRK13010
      CALL QVEC(7,13)                                                   BRK13020
      S=X(8,1)                                                          BRK13030
      CALL SUBS(13,14,S)                                                BRK13040
      S=ONEG                                                            BRK13050
      CALL MULS(14,15,S)                                                BRK13060
      CALL DIVV(15,3,16)                                                BRK13070
      CALL INTV(16,17)                                                  BRK13080
      F(8,7)=X(17,N)                                                    BRK13090
C                                                                       BRK13100
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * BRK13110
C                                                                       BRK13120
C NOW WE HAVE ALL THE COEFFICIENTS EXCEPT THOSE IN COLUMN ONE WHICH     BRK13130
C  DEPEND ON THE CHANGING VECTOR X11.  WE NOW GET THESE ELEMENTS.       BRK13140
C                                                                       BRK13150
```

```
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *  BRK13160
C                                                                        BRK13170
      CALL INTV(7,11)                                                    BRK13180
   30 CONTINUE                                                           BRK13190
C     CALL PRIV(11,2,11)                                                 BRK13200
      CALL PRIV(11,2,11)                                                 BRK13210
      CALL INTV(11,12)                                                   BRK13220
      CALL MULV(9,12,13)                                                 BRK13230
      CALL MULV(2,13,14)                                                 BRK13240
      CALL INTV(14,15)                                                   BRK13250
      D(2,1)=-ALFA(6)*X(15,N)                                            BRK13260
      CALL INTV(15,16)                                                   BRK13270
      E(4,1)=-ALFA(8)*X(16,N)                                            BRK13280
      D(4,1)=ALFA(8)*G(2)*X(12,N)                                        BRK13290
      D(6,1)=ALFA(10)*X(12,N)                                            BRK13300
      CALL QVEC(16,17)                                                   BRK13310
      F(6,1)=(ONE-ALFA(10))*X(17,N)                                      BRK13320
      D(8,1)=-X(12,N)                                                    BRK13330
      CALL MULV(9,15,18)                                                 BRK13340
      CALL SUBV(17,18,19)                                                BRK13350
      CALL DIVV(19,3,20)                                                 BRK13360
      CALL INTV(20,13)                                                   BRK13370
      F(8,1)=X(13,N)                                                     BRK13380
      CALL MULV(6,12,13)                                                 BRK13390
      CALL MULV(5,13,14)                                                 BRK13400
      CALL INTV(14,15)                                                   BRK13410
      S=X(8,N)                                                           BRK13420
      CALL MULS(12,16,S)                                                 BRK13430
      E(6,1)=(ALFA(10)-ONE)*(X(15,N)+(G(4)*X(16,N)))                     BRK13440
      CALL DIVV(15,3,17)                                                 BRK13450
      CALL INTV(17,18)                                                   BRK13460
      E(8,1)=-X(18,N)                                                    BRK13470
   29 FORMAT(//)                                                         BRK13480
      WRITE(6,29)                                                        BRK13490
      DO 500 I=1,8                                                       BRK13500
      WRITE(6,29)                                                        BRK13510
      WRITE(6,29)                                                        BRK13520
      DO 500 J=1,8                                                       BRK13530
      WRITE(6,501) D(I,J),E(I,J),F(I,J)                                  BRK13540
  500 CONTINUE                                                           BRK13550
  501 FORMAT(25X,1P3E20.8)                                               BRK13560
C     WRITE(6,29)                                                        BRK13570
C     DO 500 I=1,8                                                       BRK13580
C     WRITE(6,29)                                                        BRK13590
C       WRITE(6,29)                                                      BRK13600
C     DO 500 J=1,8                                                       BRK13610
C     WRITE(6,501) D(I,J),E(I,J),F(I,J)                                  BRK13620
C 500 CONTINUE                                                           BRK13630
```

```
C 501 FORMAT(25X,1P3E20.8)                                                    BRK13640
C                                                                             BRK13650
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *       BRK13660
C                                                                             BRK13670
C  NOW WE HAVE ALL THE COEFFICIENTS AND WILL ARRANGE THEM 16 X 16.            BRK13680
C                                                                             BRK13690
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *       BRK13700
C                                                                             BRK13710
      DO 38 I=1,16                                                            BRK13720
      DO 38 J=1,16                                                            BRK13730
      AA(I,J)=ZERC                                                            BRK13740
   38 BE(I,J)=ZERO                                                            BRK13750
      DO 40 I=1,8                                                             BRK13760
      IP=I+8                                                                  BRK13770
      AA(IP,I)=ONE                                                            BRK13780
      BB(IP,IP)=ONE                                                           BRK13790
      DC 40 J=1,8                                                             BRK13800
      JP=J+8                                                                  BRK13810
      AA(I,J)=E(I,J)                                                          BRK13820
      BB(I,J)=-F(I,J)                                                         BRK13830
   40 AA(I,JP)=D(I,J)                                                         BRK13840
C                                                                             BRK13850
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *       BRK13860
C                                                                             BRK13870
C  NOW WE SOLVE THE EIGENPROBLEM.                                             BRK13880
C                                                                             BRK13890
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *       BRK13900
C                                                                             BRK13910
CT    WRITE(6,29)                                                             BRK13920
CT    WRITE(6,29)                                                             BRK13930
      N16=16                                                                  BRK13940
      WANTX=.TRUE.                                                            BRK13950
      S=1.D-13                                                                BRK13960
      IF(KP(3).EQ.2) GO TO 401                                                BRK13970
      CALL QZ(N16,N16,BB,AA,S,NUMR,NUMI,DEN,ITER,WANTX,EKS)                   BRK13980
      GO TO 406                                                               BRK13990
  401 CALL QZ(N16,N16,AA,BB,S,NUMR,NUMI,DEN,ITER,WANTX,EKS)                   BRK14000
  406 CONTINUE                                                                BRK14010
CT    WRITE(6,29)                                                             BRK14020
CT    WRITE(6,407)                                                            BRK14030
CT407 FORMAT('0','THE FOLLOWING IS PRINT OF ITER(I).')                        BRK14040
CT    DO 110 I=1,16                                                           BRK14050
CT110 WRITE(6,100) ITER(I)                                                    BRK14060
CT100 FCRMAT(50X,I5)                                                          BRK14070
CT    WRITE(6,29)                                                             BRK14080
      DO 42 I=1,16                                                            BRK14090
      KLEE=0                                                                  BRK14100
      X(19,I)=ZERO                                                            BRK14110
```

```
      X(20,I)=ZERC                                                      BRK14120
      IF(DABS(DEN(I)).LT.1.D-20) GO TO 42                               BRK14130
      IF(DABS(NUMR(I)).LT.1.D-20) GO TO 42                              BRK14140
      IF(DABS(NUMI(I)).GT.(1.D-1*DABS(NUMR(I)))) GO TO 42               BRK14150
      IF(KP(3).EQ.1) X(19,I)=DEN(I)/NUMR(I)                             BRK14160
      IF(KP(3).EQ.2) X(19,I)=NUMR(I)/DEN(I)                             BRK14170
      IF(X(19,I).GT.ZERO) X(20,I)=ONE                                   BRK14180
      KP(9)=7                                                           BRK14190
      IF(KP(9).EQ.7) WRITE(6,41)I,X(19,I),X(20,I),NUMR(I),NUMI(I),DEN(I)BRK14200
   41 FORMAT(5X,I5,1P7E16.4)                                            BRK14210
   42 CONTINUE                                                          BRK14220
      KLEE=0                                                            BRK14230
      NVEC=0                                                            BRK14240
      S=ZERO                                                            BRK14250
      DO 43 I=1,16                                                      BRK14260
      IF(X(20,I).LT.5.D-1) GO TO 43                                     BRK14270
      IF(KLEE.EQ.0) S=X(19,I)                                           BRK14280
      IF(KLEE.EQ.0) NVEC=I                                              BRK14290
      KLEE=1                                                            BRK14300
      IF(X(19,I).LT.S) NVEC=I                                           BRK14310
      IF(X(19,I).LT.S) S=X(19,I)                                        BRK14320
   43 CONTINUE                                                          BRK14330
      RESULT(KKK)=S                                                     BRK14340
      WRITE(6,29)                                                       BRK14350
      WRITE(6,44) NVEC,S                                                BRK14360
   44 FORMAT(5X,'WE SELECT EIGVAL NUMBER ',I2,' WHICH EQUALS ',D20.5)   BRK14370
      WRITE(6,29)                                                       BRK14380
      WRITE(6,45)                                                       BRK14390
   45 FORMAT(5X,'THE CORRESPONDING EIGVEC (WE USE ONLY REAL PART) IS:') BRK14400
      WRITE(6,29)                                                       BRK14410
      DO 46 I=1,8                                                       BRK14420
      IP=I+8                                                            BRK14430
      B(I)=EKS(IP,NVEC)                                                 BRK14440
      WRITE(6,47) I,B(I)                                                BRK14450
   46 CONTINUE                                                          BRK14460
   47 FORMAT(10X,I5,2D19.6)                                             BRK14470
  403 LAM=S                                                             BRK14480
  404 CONTINUE                                                          BRK14490
      WRITE(6,412)                                                      BRK14500
  412 FORMAT('0','THE FOLLOWING IS WRITE(6,2002) CUTPUT.')              BRK14510
      WRITE(6,2002) KP(2),KKK,NVEC,LAM                                  BRK14520
 2002 FORMAT(3X,3I6,1P1E19.5)                                           BRK14530
C                                                                       BRK14540
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * BRK14550
C                                                                       BRK14560
C   NOW THE NEW PHI-DOT MUST BE CREATED FROM THE SELECTED EIGENVALUE    BRK14570
C  AND EIGENVECTOR.                                                     BRK14580
C                                                                       BRK14590
```

```
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * BRK14600
C                                                                       BRK14610
      CALL INTV(11,12)                                                  BRK14620
      CALL MULV(9,12,13)                                                BRK14630
      CALL MULV(6,12,14)                                                BRK14640
      CALL MULV(2,13,15)                                                BRK14650
      CALL MULV(5,14,16)                                                BRK14660
      CALL INTV(15,17)                                                  BRK14670
      CALL INTV(17,18)                                                  BRK14680
      CALL QVEC(18,19)                                                  BRK14690
      CALL INTV(16,20)                                                  BRK14700
      CALL MULV(9,17,12)                                                BRK14710
      CALL SUBV(19,12,13)                                               BRK14720
      S=LAM**2                                                          BRK14730
      CALL MULS(13,14,S)                                                BRK14740
      S=LAM                                                             BRK14750
      CALL MULS(20,15,S)                                                BRK14760
      CALL SUBV(14,15,16)                                               BRK14770
      S=B(1)                                                            BRK14780
      CALL MULS(16,17,S)                                                BRK14790
      CALL DUPV(17,12)                                                  BRK14800
C                                                                       BRK14810
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * BRK14820
C                                                                       BRK14830
C  VECTOR X12 NOW CONTAINS THE B1 PORTION OF PHI-DOT. THE PORTIONS      BRK14840
C WITH THE OTHER CONSTANTS WILL BE ADDEC TO VECTOR X12 AS THEY ARE      BRK14850
C DEVELOPED.                                                            BRK14860
C                                                                       BRK14870
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * BRK14880
C                                                                       BRK14890
      CALL INTV(7,13)                                                   BRK14900
      CALL MULV(9,13,14)                                                BRK14910
      CALL MULV(6,13,15)                                                BRK14920
      CALL MULV(2,14,16)                                                BRK14930
      CALL INTV(16,17)                                                  BRK14940
      CALL INTV(17,18)                                                  BRK14950
      CALL QVEC(18,19)                                                  BRK14960
      CALL MULV(9,17,20)                                                BRK14970
      CALL MULV(5,15,13)                                                BRK14980
      CALL INTV(13,14)                                                  BRK14990
      CALL SUBV(19,20,15)                                               BRK15000
      S=LAM**2                                                          BRK15010
      CALL MULS(15,16,S)                                                BRK15020
      S=LAM                                                             BRK15030
      CALL MULS(14,17,S)                                                BRK15040
      CALL SUBV(16,17,18)                                               BRK15050
      S=B(2)                                                            BRK15060
      CALL MULS(18,19,S)                                                BRK15070
```

```
      CALL ADDV(12,19,13)
      CALL DUPV(13,12)
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C  NOW B3 PORTION.
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      CALL MULV(2,9,13)
      CALL INTV(13,14)
      CALL INTV(14,15)
      CALL QVEC(15,16)
      CALL MULV(9,14,17)
      CALL SUBV(16,17,18)
      S=LAM**2
      CALL MULS(18,19,S)
      CALL MULV(5,6,13)
      CALL INTV(13,14)
      S=LAM
      CALL MULS(14,15,S)
      CALL SUBV(19,15,20)
      S=B(3)
      CALL MULS(20,19,S)
      CALL ADDV(12,19,13)
      CALL DUPV(13,12)
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C  NCW B4 PCRTION.
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      CALL INTV(7,13)
      CALL MULV(2,13,14)
      CALL INTV(14,15)
      CALL INTV(15,16)
      CALL QVEC(16,17)
      CALL MULV(9,15,18)
      CALL ADDV(18,10,19)
      CALL SUBV(19,17,20)
      S=LAM**2
      CALL MULS(20,13,S)
      S=B(4)
      CALL MULS(13,14,S)
      CALL ADDV(12,14,15)
      CALL DUPV(15,12)
C
```

```
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C  NOW B5 PORTION.
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      CALL INTV(2,13)
      CALL INTV(13,14)
      CALL QVEC(14,15)
      CALL MULV(9,13,16)
      CALL SUBV(16,15,17)
      S=LAM**2
      CALL MULS(17,18,S)
      S=B(5)
      CALL MULS(18,19,S)
      CALL ADDV(12,19,20)
      CALL DUPV(20,12)
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C  NCW B6 PORTION.
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      CALL INTV(7,13)
      CALL QVEC(13,14)
      CALL SUBV(9,14,15)
      S=LAM**2
      CALL MULS(15,16,S)
      S=B(6)
      CALL MULS(16,17,S)
      CALL ADDV(12,17,18)
      CALL DUPV(18,12)
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C   NOW B(8) PORTION (B(7) MAKES NO CONTRIBUTION TO PHI-DOT.)
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
      S=-LAM**2*B(8)
      CALL ADDS(12,12,S)
      CALL DIVV(12,3,13)
C
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * *
C
C  THE PHI-DOT VECTOR IS NOW FORMED ANC CONTAINED IN X13.
C  NOW FORM THE NEW NORMALIZED X11.
```

```
C                                                                       BRK16040
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * BRK16050
C                                                                       BRK16060
      S=X(13,1)                                                         BRK16070
      CALL SUBS(13,12,S)                                                BRK16080
      CALL INTV(12,13)                                                  BRK16090
      S=X(13,N)                                                         BRK16100
      CALL DIVS(12,11,S)                                                BRK16110
C                                                                       BRK16120
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * BRK16130
C                                                                       BRK16140
C  NOW THE NEW X11(NORMALIZED) HAS BEEN FORMED.                         BRK16150
C  CONTINUE ITERATION UNTIL SATISFACTORY CONVERGENCE IS ATTAINED.       BRK16160
C                                                                       BRK16170
C * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * BRK16180
C                                                                       BRK16190
      IF(KKK.EQ.10) GO TO 60                                            BRK16200
      KKK=KKK+1                                                         BRK16210
      GO TO 30                                                          BRK16220
   60 CONTINUE                                                          BRK16230
      WRITE(6,61) KP(2),(RESULT(J),J=6,10)                              BRK16240
   61 FORMAT(3X,1I5,1P5E20.5)                                           BRK16250
      RETURN                                                            BRK16260
      END                                                               BRK16270
      SUBROUTINE ADDV(N1,N2,N3)                                         BRK16280
C                                                                       BRK16290
      REAL*8 X(30,101),ALFA(12),G(4),S                                  BRK16300
      COMMON X,ALFA,G,N,KP                                              BRK16310
      DO 1 I=1,N                                                        BRK16320
    1 X(N3,I)=X(N1,I)+X(N2,I)                                           BRK16330
      RETURN                                                            BRK16340
      END                                                               BRK16350
      SUBROUTINE SUBV(N1,N2,N3)                                         BRK16360
C                                                                       BRK16370
      REAL*8 X(30,101),ALFA(12),G(4),S                                  BRK16380
      COMMON X,ALFA,G,N,KP                                              BRK16390
      DO 1 I=1,N                                                        BRK16400
    1 X(N3,I)=X(N1,I)-X(N2,I)                                           BRK16410
      RETURN                                                            BRK16420
      END                                                               BRK16430
      SUBROUTINE MULV(N1,N2,N3)                                         BRK16440
C                                                                       BRK16450
      REAL*8 X(30,101),ALFA(12),G(4),S                                  BRK16460
      COMMON X,ALFA,G,N,KP                                              BRK16470
      DO 1 I=1,N                                                        BRK16480
    1 X(N3,I)=X(N1,I)*X(N2,I)                                           BRK16490
      RETURN                                                            BRK16500
      END                                                               BRK16510
```

```
      SUBROUTINE DIVV(N1,N2,N3)
C
      REAL*8 X(30,101),ALFA(12),G(4),S
      COMMON X,ALFA,G,N,KP
      DO 1 I=1,N
    1 X(N3,I)=X(N1,I)/X(N2,I)
      RETURN
      END
      SUBROUTINE ADDS(N1,N2,S)
C
      REAL*8 X(30,101),ALFA(12),G(4),S
      COMMON X,ALFA,G,N,KP
      DO 1 I=1,N
    1 X(N2,I)=X(N1,I)+S
      RETURN
      END
      SUBROUTINE SUBS(N1,N2,S)
C
      REAL*8 X(30,101),ALFA(12),G(4),S
      COMMON X,ALFA,G,N,KP
      DO 1 I=1,N
    1 X(N2,I)=X(N1,I)-S
      RETURN
      END
      SUBROUTINE MULS(N1,N2,S)
C
      REAL*8 X(30,101),ALFA(12),G(4),S
      COMMON X,ALFA,G,N,KP
      DO 1 I=1,N
    1 X(N2,I)=X(N1,I)*S
      RETURN
      END
      SUBROUTINE DIVS(N1,N2,S)
C
      REAL*8 X(30,101),ALFA(12),G(4),S
      COMMON X,ALFA,G,N,KP
      DO 1 I=1,N
    1 X(N2,I)=X(N1,I)/S
      RETURN
      END
      SUBROUTINE PRIV(N1,I,J)
C
      REAL*8 X(30,101),ALFA(12),G(4),S
      COMMON X,ALFA,G,N,KP
C   IF SECOND ARGUMENT EQUALS 0, GO DIRECTLY TO RETURN
C   IF SECOND ARGUMENT EQUALS 1,PRINT THE VECTOR.
C   IF SECOND ARGUMENT EQUALS 2, PRINT THE IDENTITY AND THE VECTOR.
C   IF SECOND ARGUMENT EQUALS 3, PRINT THE VECTOR NUMBER AND THE VECTOR.
```

```
C     IF SECOND ARGUMENT EQUALS 4, PRINT NUMBER, IDENTITY, AND VECTOR.        BRK17000
C     IF SECOND ARGUMENT EQUALS 5, PRINT IDENTITY ONLY.                       BRK17010
      IF(I.EQ.0) GO TO 10                                                     BRK17020
      IF(I.EQ.1) GO TO 1                                                      BRK17030
      IF(I.EQ.2) GO TO 2                                                      BRK17040
      IF(I.EQ.3) GO TO 3                                                      BRK17050
      IF(I.EQ.4) GO TO 4                                                      BRK17060
      IF(I.EQ.5) GO TO 5                                                      BRK17070
    1 DO 8 K=1,N                                                              BRK17080
    8 WRITE(6,9) K,X(N1,K)                                                    BRK17090
    9 FORMAT(30X,I5,1PE20.5)                                                  BRK17100
   10 RETURN                                                                  BRK17110
    2 WRITE(6,21) J                                                           BRK17120
   21 FORMAT(//,30X,'VECTOR WITH IDENTITY ',I5,' FOLLOWS:')                   BRK17130
      GO TO 1                                                                 BRK17140
    3 WRITE(6,31) N1                                                          BRK17150
   31 FORMAT(//,30X,'VECTOR NUMBER ',I5,' FOLLOWS:')                          BRK17160
      GO TO 1                                                                 BRK17170
    4 WRITE(6,41) N1,J                                                        BRK17180
   41 FORMAT(//,30X,'VECTOR X',I2,' WITH IDENTITY ',I5,' FOLLOWS:')           BRK17190
      GO TO 1                                                                 BRK17200
    5 WRITE(6,51) J                                                           BRK17210
   51 FORMAT(//,30X,'VECTOR WITH IDENTITY ',I5,' HAS BEEN GENERATED.')        BRK17220
      GO TO 10                                                                BRK17230
      END                                                                     BRK17240
      SUBROUTINE INTV(N1,N2)                                                  BRK17250
      REAL*8 X(30,101),ALFA(12),G(4),S,ADD,NINO,NTNO,FIVO,THTO,EN,R           BRK17260
      COMMON X,ALFA,G,N,KP                                                    BRK17270
      EN=N-1                                                                  BRK17280
      EN=1.D+0/EN                                                             BRK17290
      NINO=EN*9.D+0/2.4D+1                                                    BRK17300
      NTNO=EN*1.9D+1/2.4D+1                                                   BRK17310
      FIVO=EN*5.D+0/2.4D+1                                                    BRK17320
      THTO=EN*1.3D+1/2.4D+1                                                   BRK17330
      R=EN/2.4D+1                                                             BRK17340
      X(N2,1)=0.D+0                                                           BRK17350
      X(N2,2)=NINO*X(N1,1)+NTNO*X(N1,2)-FIVO*X(N1,3)+X(N1,4)*R                BRK17360
      NM3=N-3                                                                 BRK17370
      DO 1 K=1,NM3                                                            BRK17380
      KP1=K+1                                                                 BRK17390
      KP2=K+2                                                                 BRK17400
      KP3=K+3                                                                 BRK17410
      ADD=THTO*(X(N1,KP1)+X(N1,KP2))-R*(X(N1,K)+X(N1,KP3))                    BRK17420
    1 X(N2,KP2)=X(N2,KP1)+ADD                                                 BRK17430
      X(N2,N)=X(N2,N-1)+NINO*X(N1,N)+NTNO*X(N1,N-1)-FIVO*X(N1,N-2)            BRK17440
     1+X(N1,N-3)*R                                                            BRK17450
      RETURN                                                                  BRK17460
      END                                                                     BRK17470
```

```
      SUBROUTINE DUPV(N1,N2)
C
      REAL*8 X(30,101),ALFA(12),G(4),S
      COMMON X,ALFA,G,N,KP
      DO 1 I=1,N
    1 X(N2,I)=X(N1,I)
      RETURN
      END
      SUBROUTINE NORV(N1)
C
      REAL*8 X(30,101),ALFA(12),G(4),S
      CCMMON X,ALFA,G,N,KP
      CALL INTV(N1,20)
      IF(DABS(X(20,N)).GT.1.D-10) GO TO 2
      DO 1 I=1,N
    1 X(19,I)=DABS(X(N1,I))
      CALL INTV(19,20)
    2 CALL DIVS(N1,18,X(20,N))
      CALL DUPV(18,N1)
      RETURN
      END
      SUBROUTINE QVEC(I,J)
C
      REAL*8 X(30,101),ALFA(12),G(4),S
      COMMON X,ALFA,G,N,KP
      CALL MULV(5,I,20)
      CALL INTV(20,19)
      CALL MULV(8,I,20)
      CALL ADDV(19,20,J)
      RETURN
      END
C
```

BIBLIOGRAPHY

1. Brock, J. E., 1978: A General Procedure for Lateral Buckling of Beams, to appear as a Naval Postgraduate School Technical Report.

2. Timoshenko, S. P. and J. M. Gere, 1961: Theory of Elastic Stability, 2nd edition, McGraw-Hill Book Company.

3. Cole, J. D., 1968: Perturbation Methods in Applied Mathematics, Blaisdell Publishing Co., Watham, Mass.

4. Moler, C. B. and G. W. Stewart, 1973: "An Algorithm for Generalized Matrix Eigenvalue Problems," SIAM Journal on Numerical Analysis, Vol. 10, No. 2, April 1973, pp. 241-256.

5. Milne, W. E., 1949: Numerical Calculus, Princeton University Press.

6. Moler, C. B. and G. W. Stewart, 1971: An Algorithm for the Generalized Matrix Eigenvalue Problem, Computer Science Department, Stanford University.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0142 Naval Postgraduate School Monterey, California 93940	2
3.	Department Chairman, Code 69 Department of Mechanical Engineering Naval Postgraduate School Monterey, California 93940	1
4.	Professor J. E. Brock, Code 69Bc Department of Mechanical Engineering Naval Postgraduate School Monterey, California 93940	1
5.	LCDR Robert E. Brown, USN 7812 Livingston Road Oxon Hill, Maryland 20021	1
6.	LCDR John Michael Fahey, USN Student Mail Center Box #1709 Naval Postgraduate School Monterey, California 93940	1

Thesis
B82237 Brown 176508
c.1
Lateral buckling of beams without warping rigidity.